

# MrRoboto

## 1 How to setup system

### 1.1 Initial computer settings

1. Install ubuntu 12.04 on your computer.
2. Install ROS GROOVY according to: <http://wiki.ros.org/groovy/Installation/Ubuntu>
3. Install git, type:  
`$ sudo apt-get install git` (in a terminal window)  
Password is *ubuntu* for the computer in the robot lab.
4. Clone the repository of the project. cd\* into a folder where you want the folder containing the code. Type:  
`$ git clone git@github.com:matni796/robot-security-tsbb11`  
(\*cd is a terminal command. If your not familiar with navigation in terminal, see [http://linuxcommand.org/lc3\\_lts0020.php](http://linuxcommand.org/lc3_lts0020.php))
5. Source your own created packages in .bashrc: type: gedit /.bashrc add the line: `source wherever_path_you_put_repository/robotsecuritytsbb11/catkin_ws/src/devel/setup.bash`
6. Install freenect, type:  
`$ sudo apt-get install ros-groovy-freenect-stack`  
`$ sudo apt-get install ros-groovy-freenect-launch`
7. Disable gspca kernel according to: [http://openkinect.org/wiki/Getting\\_Started](http://openkinect.org/wiki/Getting_Started)
8. cd into the installed folder and into catkin\_ws and type catkin\_make to build.

### 1.2 Setup of DX100 controller

1. Make sure the controller's version supports MotoPlus applications, should be a version ending with -14. Current version (20131206) is DS3.53.01A-14.
2. Load parameters. The file ALL.PRM can be found in git repo, under the folder robot/. Transfer it to a cf card or usb. Start the controller regularly. Go into management mode (see below). Then go to EX MEMORY → FOLDER, set folder where you put ALL.PRM. EX MEMORY → LOAD → PARAMETERS → BATCH PARAMETERS ALL.PRM

You might need to do a safety reset of the flash device. This is done by starting up the controller in Maintenance mode. In Maintenance Mode, enter Management mode. INITIALIZE → system flash safety reset, might take a while, wait for beep (CHECK NAME). Shut off controller and restart it regularly.

NOTE: Might change the setup of the controller. Should be done with caution. Contact Yaskawa if uncertain.

3. To install MotoPlus application, follow the tutorial on [http://wiki.ros.org/motoman\\_driver/Tutorials/In](http://wiki.ros.org/motoman_driver/Tutorials/In)

- For step 3 in tutorial, see PDF file *MotoPlus Application Installation* in folder *robot/*. Follow the instructions on step 2.1. The application file to be loaded is also in the *robot/* folder.

NOTE: We have used an .out file hardcoded for a SIA20D robot. This version does not come with the motoman files from ROS.

- For step 4 in tutorial, transfer *INIT\_ROS.JBI* to CF card or USB device. File can be found under *../catkin\_ws/src/motoman/motoman\_driver/Inform/DX100/*. Start the controller regularly. In menu, go to: Ex MEMORY → FOLDER move to the folder where you put *INIT\_ROS.JBI* file. Then: Ex Memory → LOAD → job.

### 1.3 Setup of computer network settings

1. Connect the computer to the controller via ethernet cable. Use the output CN104 on the YCP01 board.
2. Edit settings for wired/local network on computer. Set computer wired network address to 192.168.255.9 and netmask 255.255.255.0.

NOTE: On the computer, running Ubuntu 12.04, in the robot lab, there is a profile for the connection named “DX100”. When connecting the controller to the computer via ethernet, choose this.

### 1.4 Run program/Use of interface between controller and computer.

Källskript bör ordnas.

Follow the instructions on [http://wiki.ros.org/motoman\\_driver/Tutorials/Usage](http://wiki.ros.org/motoman_driver/Tutorials/Usage).

- On step 3, enter the controller IP **192.168.255.1**.
- On step 3.1.2.1, use the path *../catkin\_ws/src/motoman/motoman\_config/cfg/sia20D\_mesh.xml*
- To obtain visualization, type:  
`$ roslaunch sia20d_mesh_arm_navigation planning_scene_warehouse_viewer_sia20d_mesh.launch`
- Choose the correct frame, baselink something...
- vad mer behöver man köra. kan vi skriva ett källskript som sköter ovanstående.